8

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/666,853	09/20/2000	Shin-ichiro Tago	FUJH 17.759	5899

26304 7590 02/09/2004

KATTEN MUCHIN ZAVIS ROSENMAN
575 MADISON AVENUE
NEW YORK, NY 10022-2585

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	PAPER NUMBER
2183	8

DATE MAILED: 02/09/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

8

Office Action Summary	**Application No.** 09/666,853	**Applicant(s)** TAGO ET AL.	
	Examiner David J. Huisman	**Art Unit** 2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *31 December 2003*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *20 September 2000* is/are: a)□ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)□ Some * c)□ None of:

1.☒ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. Claims 1-18 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: #6. Extension of Time (2 months) as received on 12/31/2003 and #7. Amendment "A" as received on 12/31/2003.

Specification

3. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

Drawings

4. In general, several of the drawings are quite hard to understand given the description within the specification. Fig.6-8 and Fig.11 are very hard to understand with all of the ovals and arrows. More specifically, there are arrows pointing to other arrows and arrows pointing to dotted lines, where the significance of these dotted lines is not clear to the examiner. These figures should be illustrated more clearly for easier understanding. Appropriate changes should be made.

Claim Objections

5. Claim 1 is objected to because of the following informalities: In the first line on page 10, please replace “said first and second instruction sequence” with --said first and second instruction sequences--. Also, in line 1, insert a comma after “decodes.” Appropriate correction is required.

6. Claim 1 recites the limitation "the instruction sequence" in line 10. There is insufficient antecedent basis for this limitation in the claim because this refers to a single particular instruction sequence whereas in line 6, plural instruction sequences are buffered. The examiner recommends replacing “the instruction sequence” in line 10 with --the instruction sequences--.

7. Claim 1 recites the limitation "the branch target address information" in lines 13-14. There is insufficient antecedent basis for this limitation in the claim because this refers to data whereas in line 11, “a branch target address information buffering portion” refers to a type of memory which is used to store the data.

8. Claim 1 recites the limitation "the branch target address" in line 14 on page 9. There is insufficient antecedent basis for this limitation in the claim because this refers to the actual branch target address whereas in lines 13-14 on the same page, “a branch target address information” refers to information that is used to generate the actual branch target address instead of the actual branch target address.

9. Claim 1 recites the limitation "said first branching instruction" in the second to last line on page 9. There is insufficient antecedent basis for this limitation in the claim.

10. Claim 1 recites the limitation "said first and second instruction sequence" in the first line on page 10. There is insufficient antecedent basis for this limitation in the claim, because a second instruction sequence was never previously mentioned within the claim.

11. Claim 1 recites the limitation "said plurality of instruction buffers" in line 2 on page 10. There is insufficient antecedent basis for this limitation in the claim because the examiner is unclear as to whether the applicant is referring to a one and another one of said plurality of instruction buffers or other instruction buffers in addition to a one and other of the plurality of instruction buffers.

12. Claim 1 recites the limitation "said instruction buffer" in the last two lines of the claim. There is insufficient antecedent basis for this limitation in the claim because the applicant had previously mentioned "one of said plurality of instruction buffers" and "another one of said plurality of instruction buffers." Therefore, it is not clear what instruction buffer the applicant is referring to in the last two lines of the claim.

13. Claim 2 recites the limitation "the instruction sequence" in line 2 on page 11. There is insufficient antecedent basis for this limitation in the claim because this refers to a single particular instruction sequence whereas in line 6 of the claim (on page 10), plural instruction sequences are buffered. The examiner recommends replacing "the instruction sequence" on page 2, line 11 with --the instruction sequences--.

14. Claim 2 recites the limitation "the branch target address information" in line 5 on page 11. There is insufficient antecedent basis for this limitation in the claim because this refers to data whereas in line 3 on the same page, "a branch target address information buffering portion" refers to a type of memory which is used to store the data.

15. Claim 2 recites the limitation "the branch target address" in line 6 on page 11. There is insufficient antecedent basis for this limitation in the claim because this refers to the actual branch target address whereas in line 5 on the same page, "a branch target address information" refers to information that is used to generate the actual branch target address instead of the actual branch target address.

16. Claim 2 recites the limitation "the branch target" in line 11 on page 11. There is insufficient antecedent basis for this limitation in the claim because this could refer to the branch target address, which is mentioned previously, but it could also refer to the instruction which is located at the branch target address, which is not previously mentioned.

17. Claim 3 recites the limitation "the state" in line 2 of the claim. There is insufficient antecedent basis for this limitation in the claim. The examiner recommends replacing this with --a state--.

18. Claim 3 is objected to because of the following informalities: In lines 3 and 14 of the claim, replace "buffer" with --buffers--. Appropriate correction is required.

19. Claim 4 recites the limitation "the state" in line 2 of the claim. There is insufficient antecedent basis for this limitation in the claim. The examiner recommends replacing this with --a state--.

20. Claim 4 recites the limitation "the one of the branch target" in lines 3-4 of the claim. There is insufficient antecedent basis for this limitation in the claim.

21. Claim 4 is objected to because of the following informalities: In the last line of the claim, replace "buffer" with --buffers--. Appropriate correction is required.

22. Claim 6 recites the limitation "the instruction sequence" in line 10. There is insufficient antecedent basis for this limitation in the claim because this refers to a single particular instruction sequence whereas in line 6, plural instruction sequences are buffered. The examiner recommends replacing “the instruction sequence” in line 10 with --the instruction sequences--.

23. Claim 6 recites the limitation "the branch target address information" in line 14 of the claim. There is insufficient antecedent basis for this limitation in the claim because this refers to data whereas in line 12, “a branch target address information buffering portion” refers to a type of memory which is used to store the data.

24. Claim 6 recites the limitation "the branch target address" in line 15 of the claim. There is insufficient antecedent basis for this limitation in the claim because this refers to the actual branch target address whereas in line 14 on the same page, “a branch target address information” refers to information that is used to generate the actual branch target address instead of the actual branch target address.

25. Claim 6 recites the limitation "said first and second instruction sequence" in line 17 of the claim. There is insufficient antecedent basis for this limitation in the claim.

26. Claim 6 is objected to because of the following informalities: Replace “first and second instruction sequence” with --first and second instruction sequences--. Appropriate correction is required.

27. Claim 6 is objected to because of the following informalities: The examiner believes that the word “to” should be removed after “following” in the third to last line on page 14. Appropriate correction is required.

28. Claim 6 recites the limitation "the first branch target address information buffer" in the last two lines on page 14. There is insufficient antecedent basis for this limitation in the claim.

29. Claim 6 recites the limitation "said branch target instruction sequence" in line 2 on page 15. There is insufficient antecedent basis for this limitation in the claim.

30. Claim 6 recites the limitation "the second branch target address information buffer" in lines 3-4 on page 15. There is insufficient antecedent basis for this limitation in the claim.

31. Claim 6 recites the limitation "said instruction buffer" in lines 7-8 on page 15. There is insufficient antecedent basis for this limitation in the claim because this limitation refers to a single buffer and, no single instruction buffer had been previously singled out from the plurality of instruction buffers. Therefore, it is not clear what instruction buffer the applicant is referring to in the last two lines of the claim.

32. Claim 7 recites the limitations "said branching instruction" and "the branching instruction." There is insufficient antecedent basis for this limitation in the claim because in claim 6, the applicant refers to both a first and second branching instruction.

33. Claim 8 recites the limitations "said branching instruction" and "the branching instruction." There is insufficient antecedent basis for this limitation in the claim because in claim 6, the applicant refers to both a first and second branching instruction.

34. Claim 8 recites the limitation "the branching prediction information." There is insufficient antecedent basis for this limitation in the claim.

35. Claim 9 recites the limitations "said branching instruction" and "the branching instruction." There is insufficient antecedent basis for this limitation in the claim because in claim 6, the applicant refers to both a first and second branching instruction.

36. Claim 11 recites the limitation "the instruction buffer" in line of the claim. There is insufficient antecedent basis for this limitation in the claim because this limitation refers to a single buffer and, no single instruction buffer had been previously singled out from the plurality of instruction buffers. Therefore, it is not clear what instruction buffer the applicant is referring to in line 2 of the claim.

37. Claim 12 recites the limitation "the branching direction" in the third to last line of the claim. There is insufficient antecedent basis for this limitation in the claim.

38. Claim 16 recites the limitation "the branching direction" in the fifth to last line of the claim. There is insufficient antecedent basis for this limitation in the claim.

39. Claim 16 is objected to because of the following informalities: The end of claim 16 appears to be accidentally deleted. Applicant should insert the phrase --performs a cache miss with respect to an instruction in the determined branching direction, said cache controller performs a memory bus access.-- at the end of claim 16. Appropriate correction is required.

40. Claim 18 is objected to because of the following informalities: The examiner feels that “instruction sequential side” and “instruction target side” should be replaced with --sequential side instruction sequence-- and --target side instruction sequence--, respectively. Appropriate correction is required.

Claim Rejections - 35 USC § 112

41. Claim 1 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The first two lines on page 10 are unclear to the examiner. More specifically, it is

claimed that first and second instruction sequences are fetched from the instruction store and stored in said plurality of instruction buffers. However, the applicant had previously mentioned in claim 1 (in the last paragraph on page 9) that the first processed instruction sequence is already stored in one of said plurality of instruction buffers. Therefore, it is not clear why the first instruction sequence would already exist within one of the instruction buffers and be processed and then afterwards, fetch the same instruction sequence into an instruction buffer.

Withdrawn Rejections

42. Through amendment, applicant has overcome the rejections set forth in the previous Office Action, mailed on July 31, 2003, for claims 1 and 5-11. Consequently, these rejections are hereby withdrawn by the examiner. However, upon further consideration, a new ground(s) of rejection is made below.

Maintained Rejections

43. Applicant has failed to overcome the rejections set forth in the previous Office Action, mailed on July 31, 2003, for claims 2-4 and 12-18. Consequently, these rejections are respectfully maintained by the examiner and copied below for applicant's convenience.

Maintained Claim Rejections - 35 USC § 102

44. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –

> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

45. Claim 12 is rejected under 35 U.S.C. 102(e) as being anticipated by Shiell, U.S. Patent No. 5,864,697.

46. Referring to claim 12, Shiell has taught an information processing device comprising:

a) an instruction fetch portion which fetches both a sequential side instruction sequence and a target side instruction sequence of a branch instruction. See column 2, lines 11-31. Note that if a branch is predicted taken, the fetch portion will fetch the target side instruction sequence, whereas if the branch is predicted not-taken, then the sequential side instruction sequence will be fetched. Therefore, it can be seen that both of these sequences are fetched by the fetch unit, depending on the prediction.

b) a cache controller which fetches instructions from a cache memory or from a main memory in response to a fetch request from said instruction fetch portion. See Fig.1 and Fig.2, and note that the fetch unit (Fig.2) controls what addresses are applied to the cache for fetching instructions.

c) a memory bus access portion which accesses said main memory. See Fig.1, component 12, which is connected to a bus that is connected to main memory.

d) an instruction buffer which buffers instructions which have been fetched. See Fig.2, component 60.

e) a branching prediction portion which, prior to an execution of a branching instruction, performs a branching prediction for the branching instruction which is stored in said instruction buffer. See Fig.2, component 56.

f) if the branching direction of said branching instruction is not yet determined, said cache controller performs a memory bus access to said main memory according to a branching direction predicted by the branching prediction portion. See column 5, lines 52-54, column 7, lines 38-44, and column 7, lines 11-18. Note that when a branch prediction occurs (in this case, assume the branch is predicted taken), the target address is applied to the instruction cache in order to start fetching the target sequence of instructions. However, if there is a cache miss, slower levels of memory, including main memory, may have to be accessed in order to retrieve the desired instructions, as is known in the art. This is the inherent nature of a memory hierarchy.

Maintained Claim Rejections - 35 USC § 103

47. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

48. Claims 2-4 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell, as applied above, in view of Shintani et al., U.S. Patent No. 4,532,589 (herein referred to as Shintani), and further in view of Nakanishi, U.S. Patent No. 5,835,754.

49. Referring to claim 2, Shiell has taught an information processing device which reads, buffers, decodes and executes instructions from an instruction store portion by pipeline processing, comprising:

a) an instruction reading request portion which assigns a read address to said instruction store portion. See Fig.1 and column 6, lines 53-55, and note that the reading request portion (component 26) applies address to and reads instructions from instruction store 16i.

b) an instruction buffering portion including a plurality of instruction buffers which buffer instruction sequences read from said instruction store portion. See Fig.2 and column 7, lines 47-50, and note that the instruction buffer (component 60) comprises 16 individual instruction buffers, where each buffer is an area used to store data temporarily and deliver it at a rate different from that at which it was received.

c) an instruction execution unit which decodes and executes instructions buffered by said instruction buffering portion. See Fig.1 and note that the execution unit can comprise decode components 28, 32, and 34, which decode buffered instructions, and execution units such as components 31, 40, and 42, which execute buffered instructions after they have been decoded.

d) a branching instruction detection portion which detects a branching instruction inside the instruction sequence read from said instruction store portion. Note that it is inherent that a branch instruction will be detected. The instruction type must be known so that it can be properly executed. Furthermore, the BTB (Fig.2, component 56) detects a branch based on a fetch address. See column 7, lines 58-62.

e) a branch target address information buffering portion including a plurality of branch target address information buffers which, when said branching instruction detection portion has detected a branching instruction, buffer the branch target address information for generating the branch target address of said branching instruction. See Fig.2, component 56, and column 2, lines 11-31. Note that this table holds (buffers) target address information (among other items) when a branch is detected.

f) Shiell has not taught that a first instruction sequence being processed is stored in either one of the first or second instruction buffers and when said branching instruction detection portion detects a branching instruction inside said first instruction sequence, a second instruction sequence of the branch target is stored in the other one of the first or second instruction buffers in accordance with the branch target address information of said branching instruction. However, Shintani has taught such a concept. See Fig.1A and column 5, line 58, to column 6, line 13. Note that a main stream is stored in a first instruction buffer and, upon encountering a branch instruction, a second buffer is filled with the branch target instructions so that if the branch target path is taken, the instructions will already have been fetched from memory. This would result in saving time by not having to perform the fetch after the branch result is known. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to have a first and second instruction buffer for holding a first sequence of instructions and a target-side sequence of instructions.

g) Shiell in view of Shintani has not taught that the branch target address information of a next branching instruction inside said first instruction sequence is stored in either one of first or second branch target address information buffers. However, Nakanishi has taught a system that includes multiple branch target buffers (BTBs), wherein some branch target information is stored in a first buffer and other branch target information is stored in a second buffer based on the location of the corresponding branch instruction. See Fig.6 and column 16, lines 56-62. It should be realized that this multiple BTB system is used since multiple instructions are fetched at once and it allows for making multiple branch predictions at the same time. See Fig.2 (multiple instructions fetched from cache) and column 16, lines 33-39 (multiple predictions at once). One of ordinary skill in the art would have recognized that this multiple BTB system is applicable to Shiell in view of Shintani's system since Shiell fetches multiple instructions at once. See column 6, lines 7-16. By making more predictions in parallel as opposed to serially, time can be saved if a first branch is not taken (and the second prediction must be used). Furthermore, since all branches have a corresponding entry in one of the two BTBs, it is inherent that a next branch instruction from the first sequence would be stored in one of the first or second BTBs. As a result it would have been obvious to one of ordinary skill in the art at the time of the invention to implement multiple BTBs and have the next branch instruction information be stored in one of the first or second BTBs.

h) Shiell in view of Shintani has not taught the branch target address information of the branching instruction inside said second instruction sequence is stored in the other one of said first or second branch target address information buffers. However, as described above, Nakanishi has taught a system that includes multiple branch target buffers (BTBs), wherein some branch target information is stored in a first buffer and other branch target information is stored in a second buffer based on the location of the corresponding branch instruction. See Fig.6 and column 16, lines 56-62. It should be realized that this multiple BTB system is used since multiple instructions are fetched at once and it allows for making multiple branch predictions at the same time. See Fig.2 (multiple instructions fetched from cache) and column 16, lines 33-39 (multiple predictions at once). One of ordinary skill in the art would have recognized that this multiple BTB system is applicable to Shiell in view of Shintani's system since Shiell fetches multiple instructions at once. See column 6, lines 7-16. By making more predictions in parallel as opposed to serially, time can be saved if a first branch from the fetched instructions is not taken (and the second prediction must be used). Furthermore, since all branches have a corresponding entry in one of the two BTBs, it is conceivable that a branch instruction within the second sequence would have an address that results in its target information being stored in the other one of the first or second BTBs. As a result it would have been obvious to one of ordinary skill in the art at the time of the invention to implement multiple BTBs and have information corresponding to a branch instruction within the second sequence stored in the other one of the first or second BTBs.

50. Referring to claim 3, Shiell in view of Shintani and further in view of Nakanishi has taught an information processing device as described in claim 2. Furthermore, the first paragraph of claim 3 is rejected for the same reasons set forth in the rejection of claim 2(f), (g), and (h), above. In addition, Shintani has taught:

a) if the execution of the branching instruction inside said first instruction sequence has resulted in branching, said first instruction sequence and the branch target address information of the next branching instruction inside said first instruction sequence are invalidated. See column 5, line 58, to column 6, line 7. Note that if branching occurs, then the main instruction stream, which includes the next branch, is invalidated (no longer the main instruction stream) in favor of the target instruction stream. This is done by making the target-side buffer the main stream buffer (and vice-versa).

b) a third instruction sequence of the branch target of the branching instruction inside said second instruction sequence is stored in one of said first or second instruction buffers, in accordance with the branch target address information which have been stored in the other one of said first or second branch target address information buffers. Recall from part (a) above that the target-side buffer holding the second sequence of instructions is now the main stream buffer. Therefore, if a branch is detected within the second sequence, then its target instructions will be placed into the new target-side buffer (i.e., the original main stream buffer).

c) Finally, Nakanishi has taught that the branch target address information of the next branching instruction inside said second instruction sequence is stored in one of the first or second branch target address information buffers, and the branch target address information of the branching instruction inside said third instruction sequence is stored in the other one of said first or second branch target address information buffers. It should be realized that all branches would have a place in one of the two tables and the only difference between this portion of claim 3 and claim 2(g) and (h) above is that a second sequence and third sequence are referenced instead of a first and second sequence. Therefore, this portion of claim 3 is rejected for the same reasons set forth in the rejection of claim 2(g) and (h) above.

51. Referring to claim 4, Shiell in view of Shintani and further in view of Nakanishi has taught an information processing device as described in claim 2. Furthermore, the first paragraph of claim 4 is rejected for the same reasons set forth in the rejection of claim 2(f), (g), and (h), above. In addition, Shintani has taught:

a) if the execution of the branching instruction inside said first instruction sequence has not resulted in branching, said second instruction sequence and the branch target address information of the branching instruction inside said second instruction sequence are invalidated. See column 5, line 58, to column 6, line 18. Note that if branching does not occur, then the target instruction stream, which could include a branch, is invalidated in favor of the main instruction stream. This invalidation occurs since the target path instructions are not desired for execution.

b) a fourth instruction sequence of the branch target of the next branching instruction inside said first instruction sequence is stored in one of said first or second instruction buffers, in accordance with the branch target address information which have been stored in the other one of said first or second branch target address information buffers. Recall from part (a) main stream buffer is still supplying instructions for execution. Therefore, if a next branch is detected within the first sequence, then its target instructions will be placed into the target-side buffer based on target information concerning the next branch instruction.

c) Finally, Nakanishi has taught that the branch target address information of the next branching instruction inside said first instruction sequence is stored in one of the first or second branch target address information buffers, and the branch target address information of the branching instruction inside said fourth instruction sequence is stored in the other one of said first or second branch target address information buffers. It should be realized that all branches would have a place in one of the two tables and the only difference between this portion of claim 4 and claim 2(g) and (h) above is that a second sequence and third sequence are referenced instead of a first and second sequence. Therefore, this portion of claim 4 is rejected for the same reasons set forth in the rejection of claim 2(g) and (h) above.

52. Claims 13 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell, as applied above, in view of Shintani, as applied above, and further in view of Lee et al., Instruction Cache Fetch Policies for Speculative Execution, 1995 (herein referred to as Lee).

53. Referring to claim 13, Shiell has taught an information processing device as described in claim 12.

a) Shiell has not explicitly taught fetching a sequential-side instruction sequence and a target-side instruction sequence simultaneously. However, Shintani has taught such a concept. See Fig.1A and column 5, line 58, to column 6, line 13. Note that a main stream is stored in a first instruction buffer and, upon encountering a branch instruction, a second buffer is filled with the branch target instructions so that if the branch target path is taken, the instructions will already have been fetched from memory. This would result in saving time by not having to perform the fetch after the branch result is known. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to have a first and second instruction buffer for holding a first sequence of instructions and a target-side sequence of instructions.

b) Shiell has taught that while the branching direction of said branching instruction is not yet determined, if the cache controller has performed a cache miss with respect to an instruction in the predicted branching direction of said branching instruction, said cache controller performs the memory bus access to the main memory for an instruction fetch. See column 5, lines 52-54, column 7, lines 38-44, and column 7, lines 11-18. Note that when a branch prediction occurs (in this case, assume the branch is predicted taken), the target address is applied to the instruction cache in order to start fetching the target sequence of instructions. However, if there is a cache miss, slower levels of memory, including main memory, may have to be accessed in order to retrieve the desired instructions, as is known in the art. This is the nature of a memory hierarchy.

c) Shiell in view of Shintani has not taught that while the branching direction of said branching instruction is not yet determined, if said cache controller has performed a cache miss with respect to an instruction which is not in the predicted branching direction, said cache controller does not perform the memory bus access and stops said instruction fetch. However, as discussed in column 1 and Table 1 on page 359 of Lee, a "Decode" fetch policy has been taught which only services a cache miss if it is not for a misfetched instruction (i.e., an instruction along the non-predicted path). With such a policy, this misfetched instruction will not be retrieved from main memory (i.e., the fetch is stopped). This prevents bus blocking and cache pollution because no useful cache lines are displaced by erroneous fetches. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the teachings of Lee into the system of Shiell in view of Shintani such that if said cache controller has performed a cache miss with respect to an instruction which is not in the predicted branching direction, said cache controller does not perform the memory bus access and stops the instruction fetch.

54. Referring to claim 14, Shiell has taught an information processing device as described in claim 13.

a) Shiell has not explicitly taught fetching a sequential-side instruction sequence and a target-side instruction sequence simultaneously. However, Shintani has taught such a concept. See Fig.1A and column 5, line 58, to column 6, line 13. Note that a main stream is stored in a first instruction buffer and, upon encountering a branch instruction, a second buffer is filled with the branch target instructions so that if the branch target path is taken, the instructions will already have been fetched from memory. This would result in saving time by not having to perform the fetch after the branch result is known. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to have a first and second instruction buffer for holding a first sequence of instructions and a target-side sequence of instructions.

b) Shiell in view of Shintani has not taught that while the branching direction of said branching instruction is not yet determined and the predicted branching direction of said branching instruction is the sequential side, in the event of said cache controller performing a cache miss with respect to said target side instruction, said cache controller does not perform a memory bus access and stops said instruction fetch. However, as discussed in column 1 and Table 1 on page 359 of Lee, a "Decode" fetch policy has been taught which only services a cache miss if it is not for a misfetched instruction (i.e., an instruction along the non-predicted path). With such a policy, this misfetched instruction will not be retrieved from main memory (i.e., the fetch is stopped). This prevents bus blocking and cache pollution because no useful cache lines are displaced by erroneous fetches. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the teachings of Lee into the system of Shiell in view of Shintani such that if said cache controller has performed a cache miss with respect to said target side instruction, said cache controller does not perform a memory bus access and stops the instruction fetch.

55. Claims 15-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell as applied above, in view of Lee, as applied above.

56. Referring to claim 15, Shiell has taught an information processing device as described in claim 12. Shiell has not taught that while the branching direction of said branching instruction is not yet determined, said cache controller does not perform a memory bus access after a cache miss depending on the predicted branching direction of said branching instruction. However, as discussed in column 1 and Table 1 on page 359 of Lee, a "Decode" fetch policy has been taught

which only services a cache miss if it is not for a misfetched instruction (i.e., an instruction along the non-predicted path). With such a policy, this misfetched instruction will not be retrieved from main memory (i.e., the fetch is stopped). This prevents bus blocking and cache pollution because no useful cache lines are displaced by erroneous fetches. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the teachings of Lee into the system of Shiell such that the cache controller does not perform a memory bus access after a cache miss depending on the predicted branching direction of said branching instruction.

57. Referring to claim 16, Shiell has taught an information processing device, comprising:

a) an instruction fetch portion which fetches both a sequential side instruction sequence and a target side instruction sequence of a branching instruction. See column 2, lines 11-31. Note that if a branch is predicted taken, the fetch portion will fetch the target side instruction sequence, whereas if the branch is predicted not-taken, then the sequential side instruction sequence will be fetched. Therefore, it can be seen that both of these sequences are fetched by the fetch unit, depending on the prediction.

b) a cache controller which fetches instructions from a cache memory or from a main memory in response to a fetch request from said instruction fetch portion. See Fig.1 and Fig.2, and note that the fetch unit (Fig.2) controls what addresses are applied to the cache for fetching instructions.

c) a memory bus access portion which accesses said main memory. See Fig.1, component 12, which is connected to a bus that is connected to main memory.

d) an instruction buffer which buffers instructions which have been fetched. See Fig.2, component 60.

e) a branching prediction portion which, prior to an execution of a branching instruction, performs a branching prediction of the branching instruction which is stored in said instruction buffer. See Fig.2, component 56.

f) if said branching direction of said branching instruction has been determined and said cache controller performs a cache miss with respect to an instruction in the determined branching direction, said cache controller performs a memory bus access. See column 5, lines 52-54, column 7, lines 38-44, and column 7, lines 11-18. Note that when a branch prediction occurs (in this case, assume the branch is predicted taken), the target address is applied to the instruction cache in order to start fetching the target sequence of instructions. However, if there is a cache miss, slower levels of memory, including main memory, may have to be accessed in order to retrieve the desired instructions, as is known in the art. This is the inherent nature of a memory hierarchy.

g) Shiell has not taught that if the branching direction of said branching instruction is not yet determined and said cache controller performs a cache miss with respect to an instruction fetch, said cache controller does not perform a memory bus access and stops said instruction fetch. However, as discussed in column 1 and Table 1 on page 359 of Lee, a "Decode" fetch policy has been taught which only services a cache miss if it is not for a misfetched instruction (i.e., an instruction along the non-predicted path). With such a policy, this misfetched instruction will not be retrieved from main memory (i.e., the fetch is stopped). This prevents bus blocking and cache pollution because no useful cache lines are displaced by erroneous fetches. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the teachings of Lee into the system of Shiell such that if the branching direction of said branching instruction is not yet determined and said cache controller performs a cache miss with respect to an instruction fetch, said cache controller does not perform a memory bus access and stops the instruction fetch.

58. Referring to claim 17, Shiell in view of Lee has taught an information processing device as described in claim 16. Shiell has further taught that if the branching direction of said branching instruction is not yet determined, an instruction for which a cache hit has been made is prefetched and stored in said instruction buffer. It should be realized that before a branch's direction is determined, instructions along the predicted path will be fetched from memory. If a cache hit occurs, these instruction are brought in from the cache and placed into the instruction buffer. See column 7, lines 38-44.

59. Referring to claim 18, Shiell in view of Lee has taught an information processing device as described in claim 16. Shiell has further taught that instructions are selected from either said instruction sequential side or instruction target side in said instruction buffer depending on the branching direction of the branching prediction portion, and decoded. If a branch is predicted not-taken, for instance, sequential-side instructions will be speculatively fetched, buffered, and decoded. Likewise, if a branch is predicted taken, target-side instructions will be speculatively fetched, buffered, and decoded. Therefore, it can be seen that either sequential-side or target-side instructions will be selected from the buffer depending on the predicted branching direction.

New Claim Rejections - 35 USC § 103

60. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

61. Claims 1 and 5-11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell, as applied above, in view of Shintani, as applied above, in view of Nakanishi, as applied above, and further in view of Hara, U.S. Patent No. 5,740,415.

62. Referring to claim 1, Shiell has taught an information processing device which reads, buffers, decodes, and executes instructions from an instruction store portion by pipeline processing, comprising:

a) an instruction reading request portion which assigns a read address to said instruction store portion. See Fig.1 and column 6, lines 53-55, and note that the reading request portion (component 26) applies address to and reads instructions from instruction store 16i.

b) Shiell has not explicitly taught an instruction buffering portion including a plurality of instruction buffers which buffer instruction sequences read from said instruction store portion. However, Shintani has taught such a concept. See Fig.1A and column 5, line 58, to column 6, line 13. Note that a main stream is stored in a first instruction buffer and, upon encountering a branch instruction, a second buffer is filled with the branch target instructions so that if the branch target path is taken, the instructions will already have been fetched from memory. This would result in saving time by not having to perform the instruction fetch from memory after the branch result is known. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell to include an instruction buffering portion which includes a plurality of instruction buffers, as taught by Shintani.

c) Shiell has further taught an instruction execution unit which decodes and executes instructions buffered by said instruction buffering portion. See Fig.1 and note that the execution unit can comprise decode components 28, 32, and 34, which decode buffered instructions, and execution units such as components 31, 40, and 42, which execute buffered instructions after they have been decoded.

d) Shiell has further taught a branching instruction detection portion which detects a branching instruction inside the instruction sequence read from said instruction store portion. Note that it is inherent that a branch instruction will be detected. The instruction type must be known so that it can be properly executed. Furthermore, the BTB (Fig.2, component 56) detects a branch based on a fetch address. See column 7, lines 58-62.

e) Shiell in view of Shintani has not taught a branch target address information buffering portion including at least first and second branch target address information buffers which, when said branching instruction detection portion has detected a branching instruction, buffer the branch target address information for generating the branch target address of said branching instruction. However, Nakanishi has taught a system that includes multiple branch target address information buffers (BTBs), wherein some branch target information is stored in a first buffer and other branch target information is stored in a second buffer. See Fig.6. It should be realized that this multiple BTB system is used since multiple instructions are fetched at once and it allows for making multiple branch predictions at the same time. See Fig.2 (multiple instructions fetched from cache) and column 16, lines 33-39 (multiple predictions at once). One of ordinary skill in the art would have recognized that this multiple BTB system is applicable to Shiell in view of Shintani's system since Shiell fetches multiple instructions at once. See column 6, lines 7-16, of

Shiell. By making more predictions in parallel as opposed to serially, time can be saved if a first branch is not taken (and the second prediction must be used). As a result it would have been obvious to one of ordinary skill in the art at the time of the invention to implement multiple BTBs for buffering address information of a branching instruction.

f) Shiell in view of Shintani in view of Nakanishi, specifically Shintani, has taught that when said branching instruction detection portion detects a branching instruction in a first sequence being processed, which is stored in one of said plurality of instruction buffers, a branch target instruction sequence of said first branching instruction is stored in another one of said plurality of instruction buffers and said first and second instruction sequences are fetched from said instruction store portion and stored in said plurality of instruction buffers. See column 5, line 58, to column 6, line 10. Note that when a branch is detected in a first instruction sequence stored in a first buffer, the target stream is fetched and stored in the second buffer. Again, this is an obvious modification to Shiell because time would be saved by not having to wait until the branching direction is determined to fetch target instructions from memory.

g) Shiell in view of Shintani has not taught that when said branching instruction detection portion detects a next branching instruction following the first branching instruction in said first instruction sequence, a first branch target address information of the next branching instruction is stored in the first branch target address information buffer. However, Nakanishi has taught a system that includes multiple branch target buffers (BTBs), wherein some branch target information is stored in a first buffer and other branch target information is stored in a second buffer based on the location of the corresponding branch instruction. See Fig.6 and column 16, lines 50-62. More specifically, each instruction address is 32 bits and the lowest 2 bits are used to determine which buffer (BTB) the corresponding branch instruction's target address information is stored. For instance, the information associated with a branch instruction that is fetched from an address that ends in either 00 or 01 will be stored in the first BTB. And, the information associated with a branch instruction that is fetched from an address that ends in either 10 or 11 will be stored in the second BTB. Therefore, if a next branching instruction was fetched from an address ending in 00 or 01, then its information will be stored in the first buffer. It should be realized that this multiple BTB system is used since multiple instructions are fetched at once and it allows for making multiple branch predictions at the same time. See Fig.2 (multiple instructions fetched from cache) and column 16, lines 33-39 (multiple predictions at once). One of ordinary skill in the art would have recognized that this multiple BTB system is applicable to Shiell in view of Shintani's system since Shiell fetches multiple instructions at once. See column 6, lines 7-16 of Shiell. By making more predictions in parallel as opposed to serially, time can be saved if a first branch is not taken (and the second prediction must be used). As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to implement multiple BTBs and have the next branch instruction information be stored in the first BTB.

h) Shiell in view of Shintani has not taught that when said branching instruction detection portion detects a second branching instruction in said branch target instruction sequence, a second branch target address information of the second branching instruction is stored in the second branch target address information buffer. However, Nakanishi has taught a system that includes multiple branch target buffers (BTBs), wherein some branch target information is stored in a first buffer and other branch target information is stored in a second buffer based on the location of the corresponding branch instruction. See Fig.6 and column 16, lines 50-62. More specifically, each instruction address is 32 bits and the lowest 2 bits are used to determine which buffer (BTB) the corresponding branch instruction's target address information is stored. For instance, the information associated with a branch instruction that is fetched from an address that ends in either 00 or 01 will be stored in the first BTB. And, the information associated with a branch instruction that is fetched from an address that ends in either 10 or 11 will be stored in the second BTB. Therefore, if a next branching instruction was fetched from an address ending in 10 or 11, then its information will be stored in the second buffer. It should be realized that this multiple BTB system is used since multiple instructions are fetched at once and it allows for making multiple branch predictions at the same time. See Fig.2 (multiple instructions fetched from cache) and column 16, lines 33-39 (multiple predictions at once). One of ordinary skill in the art would have recognized that this multiple BTB system is applicable to Shiell in view of Shintani's system since Shiell fetches multiple instructions at once. See column 6, lines 7-16 of Shiell. By making more predictions in parallel as opposed to serially, time can be saved if a first branch is not taken (and the second prediction must be used). As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to implement multiple BTBs and have the second branch instruction information be stored in the second BTB.

i) Shiell in view of Shintani in view of Nakanishi has not taught that when said first branching instruction is executed, said branch target address information in either the first or second branch target address information buffer is invalidated. However, Hara has taught such a concept. More specifically, when a branch has been mispredicted (which is determined after the branch has been executed), its branch target address information in the buffer is invalidated so that an erroneous prediction is not used in the future (since it's invalid). See column 3, lines 47-54, and Fig.43. As a result, in order to prevent am erroneous prediction it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell in view of Shintani in view of Nakanishi such that branch target address information is invalidated when said first branching instruction is executed, as taught by Hara. In addition, Shintani has taught that another branch target instruction sequence starts to be fetched and stored in said instruction buffer based on the branch target address information which is not invalidated. Again, see column 5, line 58, to column 6, line 10. Note that if another branch is encountered, a new target stream will be fetched into the other instruction buffer.

63. Referring to claim 5, Shiell in view of Shintani in view of Nakanishi and further in view of Hara has taught an information processing device as described in claim 1. Shiell has further taught that in response to a single instruction read request from said instruction reading request portion, a plurality of consecutive instructions from said read address are read from said instruction store portion and buffered in said instruction buffering portion. See column 6, lines 7-16. Note that instruction data fetched from a given address can contain multiple x86 instructions, which are buffered in component 60 of Fig.2 before being decoded. More specifically, see column 7, lines 47-50 and note that an instruction address addresses 16 instructions and therefore, the buffering portion has a capacity of 16.

64. Referring to claim 6, the examiner has noted that claim 6 and claim 1 have no major differences between them. Therefore, claim 6 is rejected for the same reasons set forth in the rejection of claim 1 above.

65. Referring to claim 7, Shiell in view of Shintani in view of Nakanishi and further in view of Hara has taught an information processing device as described in claim 6. Shiell has further taught that whether said branch target address information buffering portion buffers the branch target address information of said branching instruction is determined in accordance with the branching prediction information of the branching instruction which is detected by said instruction detection portion. See column 2, lines 11-31, and note that when a branch is encountered, if no dynamic prediction information for that branch exists, then an entry will be created for it in the BTB so that target address information, among other things, can be stored. Therefore, it can be seen that the buffering portion buffers target information in accordance with branch prediction information.

66. Referring to claim 8, Shiell in view of Shintani in view of Nakanishi and further in view of Hara has taught an information processing device as described in claim 6. Shiell has further taught that whether said instruction buffering portion fetches the branch target instruction sequence of said branching instruction is determined in accordance with the branching prediction information of the branching instruction which is detected by said instruction detection portion. See column 7, lines 11-18, and note that if a branch is predicted taken, then the instruction sequence starting at the branch's target address will be fetched and buffered in order to attempt to keep the pipeline full.

67. Referring to claim 9, Shiell in view of Shintani in view of Nakanishi and further in view of Hara has taught an information processing device as described in claim 6. Shiell has further taught if said branching instruction detection portion predicts with a prescribed high level of probability that the branching instruction will not branch, said branch target address information

buffering portion does not fetch the branch target instruction sequence of said branching instruction. See column 2, lines 11-31 (specifically, lines 18-21), and note that when a branch is predicted not-taken, the target address is not fetched by the BTB. Instead, the instruction at the next sequential address is fetched. The certainty of the prediction is based on a group of history bits within the BTB that provide information regarding the last "X" encounters of that particular branch. See column 8, lines 47-51. If, the branch has been taken "X" times in a row, then the BTB will generally predict with high probability that the branch will be taken again.

68. Referring to claim 10, Shiell in view of Shintani in view of Nakanishi and further in view of Hara has taught an information processing device as described in claim 6. Shiell has not taught that when said branch target address information buffering portion has buffered branch target address information of a first branching instruction, if said branching instruction detection portion has detected a second branching instruction which has a greater possibility of branching than said first branching instruction, said branch target address information buffering portion invalidates the branch target address information of said first branching instruction and buffers the branch target address information of said second branching instruction. However, Hara has taught such a concept. See column 9, lines 4-10. As disclosed by Hara, by storing a branch instruction having a high branch probability as opposed to an instruction with a small branch probability, branch prediction accuracy is improved. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Shiell's branch target address information buffer so that it operates in a manner equal to that of Hara's.

69. Referring to claim 11, Shiell in view of Shintani in view of Nakanishi and further in view of Hara has taught an information processing device as described in claim 6. Shiell has further

taught that when the instruction buffer of said instruction buffering portion is empty (it should be noted that whether the instruction buffer is empty or not, the following still hold true):

a) if a first branching instruction having a first branching possibility is detected by said branching instruction detection portion, a branch target instruction sequence of said first branching instruction is not fetched to said instruction buffering portion and said branch target address information buffering portion buffers the branch target address information of the first branching instruction. If the first branch instruction has a very low branching possibility and it is predicted not-taken by the BTB, then the target instruction sequence for that branch will not be fetched and buffered. Instead, the next sequential instruction is fetched. See column 2, lines 11-31 (specifically, lines 18-21). Regardless, the branch target address information buffering portion (BTB) will buffer the branch target address information such that the target address will be buffered along with the history update based on the outcome of the branch. See column 2, lines 15-22.

b) if said branching instruction detection portion has detected a second branching instruction which has a second branching possibility which is higher than said first branching possibility, a branch target instruction sequence of said second branching instruction is fetched to said instruction buffering portion. If the second branch instruction has a very high branching possibility and it is predicted taken by the BTB, then the target instruction sequence for that branch will be fetched and buffered. See column 2, lines 11-31 (specifically, lines 18-21).

Response to Arguments

70. Applicant's arguments filed on December 31, 2003, have been fully considered but they are not persuasive.

71. In the remarks, Applicant argues the novelty/rejection of claim 2 on pages 22-23 of the remarks, in substance that:

> "In Nakanishi, there are two BTBs (11 and 21). However, these two BTBs do not correspond to the branch instruction in the first sequence being processed and the branch instruction in the second instruction sequence, like the present invention. Nakanishi simply discloses a method for predicting the branch direction of the fetched instruction."

72. These arguments are not found persuasive for the following reasons:

a) It should be realized from column 16, lines 50-62 of Nakanishi that branch target address information of a branching instruction is stored in either the first or second BTB based on the address of the branch instruction. Each instruction address is 32 bits and the lowest 2 bits are used to determine which buffer (BTB) the corresponding branch instruction's target address information is stored. For instance, the information associated with a branch instruction that is fetched from an address that ends in either 00 or 01 will be stored in the first BTB. And, the information associated with a branch instruction that is fetched from an address that ends in either 10 or 11 will be stored in the second BTB. Therefore, when a first branch instruction's address in Nakanishi ends in either 00 or 01 and a second branch instruction's address ends in 10 or 11, Nakanishi reads on applicant's claim 2. More specifically, Nakanishi would read on applicant's claim 2 when applicant's first instruction sequence has a branch at an address ending in 00 or 01 and when applicant's second instruction sequence has a branch at an address ending in 10 or 11.

73. In the remarks, Applicant argues the novelty/rejection of claims 12 and 16 on page 25 of the remarks, in substance that:

> "According to the invention of these claims (12 and 16), when prefetching instructions, if there are cache misses via cache access, whether an external memory bus access for prefetch is performed or not is determined in accordance with the predicted branching direction. This is because the external memory bus access needs lengthy instruction cycles, therefore, if the prefetched instruction is not in the predicted branching direction, such external memory bus access is prohibited. In the Office Action (paragraph 29f), it's alleged that the external memory access when a cache miss occurs is the inherent nature of the memory hierarchy. Applicant respectfully disagrees with this conclusion. That is, in the present invention, the instruction pre-fetch to the cache memory is performed no matter what the branching prediction is, however, if a cache miss occurred, then whether the external memory access is performed is determined in accordance with the branching prediction."

74. These arguments are not found persuasive for the following reasons:

a) The applicant is reading the limitation in question too broadly. For instance, in claim 12, the limitation "if the branching direction of said branching instruction is not yet determined, said cache controller performs a memory bus access to said main memory according to a branching direction predicted by the branching prediction portion" is not limited to just prohibiting external memory bus access if the prefetched instruction is not in the predicted branching direction, as applicant is arguing. The examiner has asserted in the previous office action that this is how a memory hierarchy works. In general, when a cache miss occurs, the next slower level of memory must be accessed (in this case main memory). And, the claim merely states that a memory bus access is performed according to a predicted branching direction. This is interpreted as saying, when a branch is predicted taken, the system will try to fetch a target instruction sequence from cache. If there is a cache miss, the target instruction sequence will be fetched from main memory. On the other hand, when a branch is predicted not-taken, the system will try to fetch a sequential instruction sequence from cache. If there is a cache miss, the sequential instruction sequence will be fetched from main memory. Therefore, these different types of memory bus access are based on the predicted branching direction.

Conclusion

75. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (703) 305-7811. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
February 5, 2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100